

Overview of Phased Implementation

For most of our customers, the [ProductName] portal is only one element in their long-term IT strategy. In their environments, as in a typical enterprise, there is a generally nascent technological maturity of various systems, including Web-associated applications and tools, as well as an overall unevenness to the operational readiness of various stakeholders and teams to upgrade to cutting-edge software solutions.

Besides the technology hurdles, there are also often internal customer battles to overcome. By the time [Company] PS arrives on the scene, some camps are eager (sometimes over-eager) to show the value of their portal investment to executives and investors; other teams, for political or bureaucratic reasons, are unaware of our technology or unwilling to even consider participating in the deployment of a [ProductName] solution.

To acknowledge this reality and still provide the maximum return to our customers on their investment in our product and services, the PS team regularly breaks down [ProductName] implementations into logical phases. Doing so increases the likelihood of customer success by providing the following solution-focused benefits:

- **Rapid time-to-value for high-priority customer needs.**
Customer-side project owners are able show the value of the portal to stakeholders.
- **Near-term demonstration of portal functions and project success.**
[Company] earns widespread customer trust by making the portal valuable and visible early on in the customer engagement.
- **Maximum consulting efficiency.**
Small, focused teams work closely to provide rapid turn-around of customer requirements.
- **Realistic position of [ProductName] in an ever-changing enterprise landscape.**
Phases make it possible to situate the portal in a current business, technology, or operational reality, and then adapt as these climates evolve or transition over time.

Phased implementation is a simple concept: Divide requirements into logical phases of implementation, and treat each phase as a "mini-engagement" unto itself. This means that each phase

Obviously, the choice of what to include in a development phase is not an arbitrary decision. The PS project manager must work closely with the appropriate customer-side decision-makers to plan, schedule, and implement phases in a manner that is realistic and beneficial.

In general, the following project factors inform what gets included in the various phases of development:

- Hard deadlines for the customer to deploy the portal to some user base.
- Ability of the customer to pay for all of the features desired by a specific date.
- Readiness of the customer to make decisions about feature and function priorities.
- Readiness of the customer to provide dependent technologies to support portal functions.
- Readiness of specific portal user groups to be trained on and start using [ProductName].

BEST PRACTICE: For every requirement gathered by the PS business analyst during a Portal Workshop, test the viability of the requirement according to the above checklist of factors. If the requirement is a "slam dunk" (easy to provide) and will show immediate value to the portal users who will see it during the initial [ProductName] roll-out phase, go ahead and include it for that phase. If work is required but it will still show value, propose this to the customer along with time estimates and cost for completion. If any of the above factors are at risk for initial deployment, make a list of dependencies to discuss with the customer, and ask them to consider the requirement for a later phase.

Effect of Phases on the PS Project Team

Executing a phased implementation optimizes the team size that PS typically assembles for customer projects: a small, focused team of consultants who manage time in short increments in order to meet several short-term deadlines in succession. This practice also leverages the most use of our rapid development methodology (see "Tools and Concepts" in Chapter 2)—in particular, managing engagements and project phases to repeatable documentation (see "Document-Based Project Management" in Chapter 4).

The success of a phased implementation strategy depends on the following skills and best practices from all members of the PS project team:

- **Project Managers**

The usual rules for customer communication are more important than ever. For the initial (Pilot) phase, maintain close contact with the customer-side project champions, beginning as soon as possible after the sale of professional services is finalized.

During this time, communication should focus on carefully managing customer expectations—including any curve balls that might have been thrown or fielded during the sales cycle—and clearing any roadblocks to implementation: hardware procurement, political battles over business goals of the portal, programming of customer-side dependent systems, user training, and so on.

Following the Pilot phase, the focus of customer communication becomes focused on resolving issues that arise during acceptance and roll-out of the solution, and documenting and pursuing any opportunity for follow-on sales.

"Issue Tracking Process" in Chapter 6 contains guidelines for communication about any bugs, feature requests, or new work that customers might request following the initial roll-out of their [ProductName] solution. "Generating Follow-On Sales" later in this chapter contains tips for using our change management process to [Company]'s profit advantage.

- **Business Analysts**

In addition to closely monitoring customer expectations along with the Project Manager (see above), a Business Analyst can contribute to the success of a phased implementation by carefully planning and executing the customer's Portal Workshop. During the agenda planning and initial customer introductions (pre-Workshop), the Business Analyst should seek and record all indicators of the solution's business requirements, as the customer sees them.

For example, the customer-side portal owner tasked with rolling-out the [ProductName] solution will often say, "We want to go live with [some users] by [some date]." Already, this shows that the customer is willing and able to consider the project in terms of phases. The Business Analyst should then track down what this indication really means: Who are the users they're talking about? What are their jobs? How much do they know about portals, and how do they expect to use them? To an extent, these questions can be asked in early customer conversations, but the most helpful answers will come during the Portal Workshop itself.

At the Portal Workshop, the Business Analyst should take care to address concerns that arise and judge them according to their priority: Is the comment/request/requirement pertinent for the near-term phase, or is it an idea that is best address after the Pilot has been rolled-out? If it is best left for a later phase, what information is needed before the requirement can be made a reality? Does the portal owner even consider this requirement a priority at all?

Lastly, the Business Analyst should work closely with the Project Manager and Business Writer to make sure that all phasing of requirements raised during the Portal Workshop is clearly defined and explained in the Portal Workshop Report.

- **Developers**

Typically, developers who finish work on one phase of an engagement will branch the code tree for the project prior to starting work on the next phase. Other phased implementation best practices for PS developers include regular status meetings and code reviews with other developers, as well as in some cases code collaboration ("extreme programming"), although this latter practice has not been used much in our group to date.

Stages of a [Company] PS Engagement

Somewhere in our development as a PS organization, we adopted an idea called "the four Ds" which describes the rough stages of a custom software engagement: Discover, Design, Develop, Deliver. This high-level description of our process made for a catchy and mostly accurate shorthand for telling our engagement story to customers, to investors, to partners, and to one another around the company.

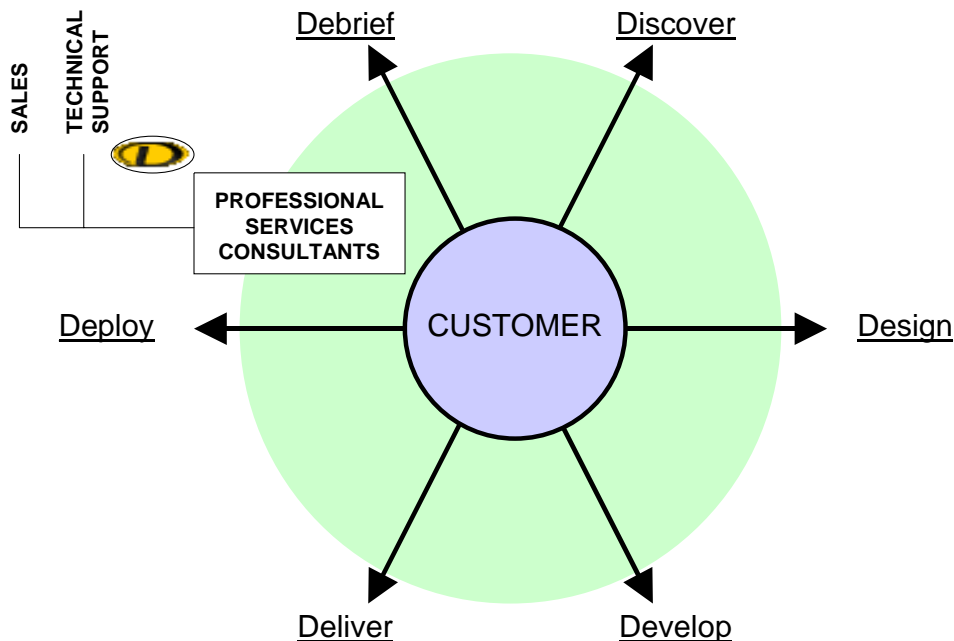
But over time we realized that these four terms lacked some of the precision necessary to describe a few other high-priority activities that must take place within each engagement time frame: specifically, the need to communicate carefully up-front among key team members so that customer expectations are accurately upheld from the sales cycle into the engagement, and the need to focus on special activities that occur at the customer site around the time their solution goes "live".

Conveniently (or perhaps maddeningly?) we can phrase these additional stages in terms that begin with "D" as well. As a result, we have the "Six Ds" of a PS engagement cycle:

- Debrief
- Discover
- Design
- Develop
- Deliver
- Deploy

The Six Ds Demystified

More than a device for remembering engagement phases, the six Ds actually provide an important tool for separating and clarifying the activities required for implementing a solution, and the identities and tasks of the roleplayers involved.



The six "Ds" of an engagement aren't stages in isolation. Often, knowledge of the details of a particular stage is required far in advance of its position in the engagement flow. For this reason, communication for the whole engagement must be centralized with a group of skilled and empowered team players, both on the customer side and among the consultants.